



Общество с ограниченной ответственностью «Ребреин»

ИНН 7727409582, ОГРН 1197746106161

Адрес: 123056, город Москва, Большая Грузинская ул, д. 36а стр. 5а, офис 13

Утверждено

Приказом № ПР-1 от 17.06.2025 г.

Генеральный директор

 Фролкина Е.А.  
«17» июня 2025 г.

ДОПОЛНИТЕЛЬНАЯ ПРОФЕССИОНАЛЬНАЯ ПРОГРАММА  
– ПРОГРАММА ПОВЫШЕНИЯ КВАЛИФИКАЦИИ

«СТАРТ В ЯЗЫКЕ ПРОГРАММИРОВАНИЯ GO:  
СИНТАКСИС, ПРАКТИКА, ПРОЕКТЫ»

**Срок реализации:** 2 недели (14 дней)

**Количество часов:** 30 акад. ч.

**Форма обучения:** заочная форма

**Формат обучения:** с применением  
исключительно дистанционных технологий

**Возраст обучающихся:** для лиц старше 17  
лет, имеющих или получающих среднее  
профессиональное и (или) высшее  
образование

Москва, 2025 г.

## **1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

Настоящая дополнительная профессиональная программа – программа повышения квалификации «Старт в языке программирования Go: синтаксис, практика, проекты» (далее – Программа) разработана в соответствии с:

- Федеральным законом от 29 декабря 2012 г. № 273-ФЗ «Об образовании в Российской Федерации»;
- Приказом Министерства образования и науки РФ от 1 июля 2013 г. № 499 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным профессиональным программам»;
- Постановлением Правительства РФ от 11.10.2023 № 1678 «Об утверждении Правил применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ»;
- Профессиональным стандартом 06.001 «Программист», утвержден приказом Министерства труда и социальной защиты Российской Федерации от 20 июля 2022 г. № 424н;
- ФГОС высшего образования – бакалавриат по направлению 09.03.04 Программная инженерия, утв. приказом Минобрнауки России от 19.09.2017 № 920;
- Локальными нормативными актами ООО «Ребреин».

В данной программе учтены основные идеи формирования универсальных учебных действий учащихся и соблюдена преемственность с программами высшего и/или среднего профессионального образования.

### **Направленность программы:**

Программа имеет техническую направленность.

### **Адресат:**

Программа предназначена для:

- Начинающих программистов, которые хотят освоить популярный язык с минимальным порогом входа и получить базу для старта карьеры в разработке
- Разработчиков с опытом на других языках программирования (например, Python, Java, C++), стремящихся освоить Go как современный инструмент для создания масштабируемых, конкурентных и облачных решений
- IT-специалистов (в том числе DevOps-инженеров и системных администраторов), заинтересованных в применении Go для написания утилит, автоматизации процессов и повышения квалификации
- Студентов и выпускников технических направлений, которым важно освоить перспективный язык с большим спросом на рынке труда

### **Требования к входным знаниям обучающегося:**

Для успешного освоения программы участникам рекомендуется:

- Иметь базовые навыки работы с компьютером и операционной системой, предпочтительно Linux или macOS
- Понимать основы алгоритмизации и логики программирования (что такое переменная, условие, цикл и т.п.)
- Иметь начальные знания любой современной системы контроля версий (например, Git) — на уровне понимания, как организована работа с кодом
- Знание любого другого языка программирования (Python, JavaScript, Java и др.) будет преимуществом, но не является обязательным
- Владение техническим английским (на уровне чтения документации) приветствуется.

### **Актуальность реализации:**

Язык программирования Go (Golang), разработанный компанией Google, сегодня широко применяется в разработке высоконагруженных сервисов, микросервисной архитектуре, системах распределённой обработки данных и инструментах DevOps. Благодаря простому синтаксису, высокой скорости выполнения и удобным средствам параллелизма Go стал одним из ключевых языков современной ИТ-индустрии.

Освоение Go открывает новые карьерные возможности для начинающих разработчиков, специалистов из смежных областей и опытных программистов, стремящихся расширить стек технологий и перейти в перспективные направления разработки. Программа даёт системное понимание базового синтаксиса, принципов языка и позволяет сформировать прочный фундамент для дальнейшего углубления в экосистему Go.

### **Отличительные особенности программы:**

- Практикоориентированный формат: каждое занятие включает практические задания, направленные на закрепление новых конструкций Go.
- Пошаговый вход в язык без перегрузки теорией — от простого к сложному, с учётом потребностей начинающих.
- Освещены ключевые особенности языка: указатели, горутины, каналы, интерфейсы, кастомные типы.
- Формирование навыков, необходимых для перехода к реальным проектам и микросервисной разработке.
- Актуальные кейсы и примеры — с ориентацией на реальные задачи backend-разработки и DevOps-инфраструктуры.
- Возможность использовать полученные знания в проектах разного уровня: от скриптов до сложных распределённых систем.

**Объем и срок освоения программы:** 30 академ. ч. в течение 2 недель (14 дней).

Доступ к материалам Программы у обучающихся остаётся и после окончания периода обучения. Это позволяет повторять изученный материал в удобное время, восполнять пробелы в знаниях, а также возвращаться к практическим заданиям при решении рабочих задач. Такой формат способствует более глубокому закреплению навыков и поддерживает профессиональное развитие выпускников даже после завершения обучения.

**Выдаваемый документ о квалификации:** удостоверение о повышении квалификации и/или сертификат об успешном освоении программы.

### **Цели и задачи программы:**

Сформировать у обучающихся базовые знания и практические навыки разработки на языке программирования Go, необходимые для выполнения прикладных и системных задач в области программной инженерии, а также обеспечить уверенное владение инструментами Go как стартовой платформой для дальнейшего профессионального роста.

### **Программа направлена на решение следующих основных задач:**

Обучающие:

- Ознакомить с синтаксисом языка Go и его основными конструкциями.
- Сформировать представление о базовых и составных типах данных в Go.
- Научить использовать механизмы обработки ошибок и работы с памятью.
- Ввести в принципы конкурентного программирования на Go (горутины, каналы).
- Сформировать понимание интерфейсов, композиции и механизмов синхронизации.

**Развивающие:**

- Развить логическое и алгоритмическое мышление через решение задач на Go
- Стимулировать способность к самообучению и анализу новых языковых конструкций
- Способствовать формированию опыта написания чистого и читаемого кода
- Развить умения проектировать простые программы и отлаживать их выполнение

**Воспитательные:**

- Воспитывать внимательность к деталям и культуру программирования
- Формировать ответственное отношение к выполнению технических заданий
- Способствовать развитию самостоятельности и настойчивости в обучении новым технологиям
- Поддерживать интерес к исследовательской и инженерной деятельности в области разработки ПО

**Планируемые результаты:****Знания:**

В результате обучения слушатель будет знать:

- Основной синтаксис языка Go, структуру программы и правила компиляции
- Примитивные и составные типы данных: массивы, слайсы, map, строки, указатели
- Принципы обработки ошибок и их типы в Go
- Основы объектно-ориентированного проектирования в стиле Go (структуры, интерфейсы, композиция)
- Принципы конкурентного программирования: горутины, каналы, race conditions
- Средства синхронизации и безопасной работы с памятью
- Инструменты Go для написания, тестирования и отладки программ

**Умения:**

В результате обучения слушатель будет уметь:

- Писать и запускать программы на Go, использовать основные конструкции языка
- Создавать и инициализировать переменные различных типов
- Обрабатывать ошибки с помощью встроенных и пользовательских механизмов
- Проектировать структуры данных и интерфейсы, обеспечивая читаемость и масштабируемость кода
- Использовать горутины и каналы для реализации параллельных задач
- Выполнять отладку, анализ производительности и устранение состояния гонки
- Работать с внешними пакетами и базовыми инструментами go mod, go run, go build, go test

**Навыки:**

В результате обучения слушатель приобретет навыки:

- Быстрой ориентации в синтаксисе и логике языка Go
- Решения практических задач с использованием idiomatic Go
- Конструирования небольших консольных приложений и модулей
- Организации безопасной и эффективной работы с параллельными процессами
- Поддержания читаемости и расширяемости кода
- Использования среды разработки и инструментов Go в повседневной практике

**Перечень профессиональных компетенций, на получение которых направлено обучение:**

На основе профстандарта 06.001 «Программист»:

- А/02.3 Написание программного кода с использованием языков программирования, определения и манипулирования данными в базах данных
- А/05.3 Проверка и отладка программного кода
- В/03.4 Проверка работоспособности компьютерного программного обеспечения

Таким образом, в результате освоения программы у обучающихся формируются следующие профессиональные компетенции:

- ОПК-2. Способен использовать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности;
- ОПК-5. Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем;
- ОПК-6. Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов;

### **Организационно-педагогические условия реализации программы дополнительного профессионального образования**

**Язык реализации образовательной программы:** обучение проводится на русском языке.

**Форма обучения:** заочная форма.

**Особенности реализации программы:** программа реализуется с использованием электронного обучения и исключительно дистанционных образовательных технологий.

**Условия набора:** на обучение принимаются все желающие лица, оплатившие обучение и заключившие договор об образовании. Обучение проходит в индивидуальном формате без формирования учебных групп. Обучающийся самостоятельно определяет время освоения Программы.

#### **Формы проведения занятий:**

- занятия в текстовом формате;
- практическая работа;
- самостоятельная работа с литературой;
- индивидуальные вопросы.

### **Материально-техническое оснащение**

#### **Материальное обеспечение программы**

Занятия проводятся в системе дистанционного обучения «Rebrain». Каждый обучающийся и педагог оснащены доступом к системе дистанционного обучения: <https://rebrainme.com/>.

У педагога дополнительного профессионального образования имеется необходимое оборудование средства для реализации программы: ноутбук с подключением к интернету, программное обеспечение.

#### **Методическое обеспечение программы**

Программа обеспечена:

- учебно-методическими материалами (текстовые занятия, полезными материалами);
- практическими заданиями.

**Кадровое обеспечение:**

К реализации программы в качестве педагогов дополнительного образования допускаются лица:

1) отвечающее одному из требований:

а) имеющее высшее образование или среднее профессиональное образование в рамках укрупненных групп специальностей и направлений подготовки высшего образования и специальностей среднего профессионального образования «Образование и педагогические науки»;

б) имеющее высшее образование либо среднее профессиональное образование в рамках иных укрупненных групп специальностей и направлений подготовки высшего образования и специальностей среднего профессионального образования при условии его соответствия дополнительной общеобразовательной общеразвивающей программе, реализуемой ООО «Ребреин», и получение при необходимости дополнительного профессионального образования педагогической направленности;

в) успешно прошедшее промежуточной аттестации не менее чем за два года обучения по образовательным программам высшего образования по специальностям и направлениям подготовки, соответствующей направленности дополнительной общеобразовательной общеразвивающей программе;

2) не имеющее ограничений на занятие педагогической деятельностью, установленных законодательством Российской Федерации;

3) прошедшее обязательный предварительный (при поступлении на работу) и периодические медицинские осмотры (обследования), а также внеочередные медицинские осмотры (обследования) в порядке, установленном законодательством Российской Федерации.

Реализация Программы также возможна лицами, привлекаемыми на условиях гражданско-правового договора в соответствии с действующим законодательством РФ.

## **2. УЧЕБНЫЙ ПЛАН**

№ п/ п	Наименование модуля	Количество часов			Формы контроля / аттестация
		Всего	Теория	Практика	
1	Модуль 1 “Онбординг”	1	0,5	0,5	Входное тестирование
2	Модуль 2 “Golang Start”	25	10	15	Практическое задание
3	Итоговая аттестация	4		4	Итоговое практическое задание

## **3. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК**

№ п/п	Наименование модуля	1 неделя	2 неделя
1	Модуль 1 “Онбординг”	1	
2	Модуль 2 “Golang Start”	14	11
3	Итоговая аттестация		4   А

## **4. РАБОЧАЯ ПРОГРАММА**

### **Модуль 1. Онбординг**

Теория 0,5 академ. ч. Практика 0,5 академ. ч.

Модуль состоит из следующих тем:

#### **Тема 1: Онбординг**

В модуле обучающемуся предоставляется вводный конспект, содержащий общую информацию о программе, структуре курса, форматах взаимодействия с материалами и ожидаемых результатах обучения.

Предусмотрено прохождение входного тестирования, включающего 7 вопросов, направленных на закрепление информации из онбординга. В рамках блока обучающийся выполняет задание по целеполаганию: формулирует свою цель прохождения программы, указывает желаемые навыки по окончании обучения, а также оценивает текущий уровень своих знаний.

### **Модуль 2. Golang Start**

Теория 10 академ. ч. Практика 15 академ. ч.

Модуль состоит из следующих тем:

#### **Тема 1: Основы синтаксиса.**

Содержание: Примитивы. Нестандартные типы. Арифметические и логические операции. Битовые операции. Условные операторы. Оператор цикла. Функции. Практическое задание.

#### **Тема 2: Указатели (Pointers).**

Содержание: Что делают указатели. Разадресация. Изменение значений через указатели. Практическое задание.

#### **Тема 3: Массивы, слайсы и тар.**

Содержание: Массивы (Инициализация с var, индексация в массиве, длина массива, матрицы из массивов). Слайсы (инициализация, добавление данных, функция make, capacity, передача параметров в функции). Мар. Практическое задание.

#### **Тема 4: Строки и байты.**

Содержание: Тип Byte. Символы для строк. Тип rune. Пакет strings. Практическое задание.

#### **Тема 5: Обработка ошибок.**

Содержание: Два способа создать ошибку. Обработка ошибок. Практическое задание.

#### **Тема 6: Кастомные типы, структуры и методы.**

Содержание: Структуры и как они работают. Кастомные типы на основе базовых. Методы: функции для типов. Практическое задание.

#### **Тема 7: Композиция и интерфейсы.**

Содержание: Композиция - определение и примеры. Интерфейсы - определение и примеры. Как их использовать. Практическое задание.

#### **Тема 8: Пустой интерфейс.**

Содержание: Что такое пустой интерфейс и как с ним работать. Слайсы и пустые интерфейсы. Практическое задание.

**Тема 9: Горутины и каналы.**

Содержание: Что такое горутина. Каналы. Что выбрать. Управление горутинами с помощью контекстов. Практическое задание.

**Тема 10: Состояние гонки и механизмы синхронизации.**

Содержание: Состояние гонки и как его обнаружить. Механизмы синхронизации. Практическое задание.

Модуль направлен на формирование у обучающихся базовых навыков программирования на языке Golang. Каждый блок модуля включает текстовое занятие с теоретическим материалом и пошаговыми инструкциями, после изучения которого предлагается практическое задание.

Практические задания рассчитаны на 2 академических часа. Выполнение заданий предполагает отправку решения на проверку через личный кабинет обучающегося. Критерии оценки прописаны в описании к каждому заданию. В случае корректного выполнения выставляется зачёт. Если работа содержит ошибки, задание возвращается на доработку. При повторной неудачной попытке (после двух доработок) обучающийся получает «незачёт».

**Итоговая аттестация.** Блок посвящён выполнению финального практического задания без предварительного теоретического блока.

## **5. МЕТОДИЧЕСКИЕ И ОЦЕНОЧНЫЕ МАТЕРИАЛЫ**

Программа обеспечена системой дистанционного обучения <https://rebrainme.com/>.

Педагогические технологии:

- технология дифференцированного обучения;
- технология разноуровневого обучения;
- технология развивающего обучения;
- технология проблемного обучения;
- технология дистанционного обучения.

Методы обучения:

- словесный, наглядный практический;
- объяснительно – иллюстративный;
- частично-поисковый, исследовательский проблемный;
- игровой, дискуссионный.

**Электронно-библиотечные ресурсы и системы, информационно-справочные системы:**

1. Научная электронная библиотека eLIBRARY.RU.
2. Собственные учебные материалы: <https://rebrainme.com/golang-s-nulya/>
3. GO Fundamentals [Электронный ресурс]: <https://go101.org/article/101.html>
4. Golang Блог о языке программирования Go [Электронный ресурс]: <https://golang-blog.blogspot.com/>
5. Изучение языка программирования Golang [Электронный ресурс]: <https://golangify.com/>

## **6. ОЦЕНКА КАЧЕСТВА ОСВОЕНИЯ ПРОГРАММЫ**

## **Оценочные материалы:**

Для отслеживания результатов освоения программы среди слушателей проводится текущий контроль и итоговое оценивание.

## **Текущий контроль**

Осуществление текущего контроля проводится после занятий в виде написания практических заданий или тестирований. Тематика и условия выполнения практических заданий расписаны в личном кабинете обучающегося в СДО. Педагог проверяет решение и принимает решение о принятии решения (зачет), о необходимости доработать решение или о незачете. Если промежуточный контроль представлен в виде тестирования, подсчет верных ответов и выставление оценки «зачёт» и «незачёт» происходят в автоматическим решиме в СДО.

Пример.

## **Тема 2: Указатели (Pointers).**

В этом задании предлагаем вам решить алгоритмическую задачку, с помощью теории из этого блока.

Вам нужно развернуть число x. Следовательно, если на входе у функции число 123 тогда на выходе должно быть число 321, а вот ещё несколько примеров, чтобы лучше понимать, что нужно сделать.

- in = 123, out = 321
- in = 87634, out = 43678
- in = 4444, out = 4444
- in = -4321, out = -1234 — знак минуса не инвертируем.
- in 2147483648, out = 0 — при условии запуска на 32 битной платформе, если перевернуть число, получится переполнение int32.
- in 8223372036854775829, out = 0 — при условии запуска на 64 битной платформе, если перевернуть это число, получится переполнение int64.

Сигнатура функции выглядит так:

```
func ReverseInt(num int) int {  
    // ... your code  
}
```

P.s. Для успешного выполнения задания необходимо поместить код решения в /home/user/reverse\_int.go

## **Тема 5: Обработка ошибок.**

1. Создайте файл .go со следующим содержимым:

```
package mini5  
  
func StringToInt64(str string) (int64, error) {  
    panic("implement me")  
}
```

2. Реализуйте функцию перевода строки в int64. Если перевести строку в число невозможно, верните пустую ошибку.

3. Текст ошибки должен содержать строку которую не возможно перевести, и сообщение о невозможности перевода.

4. Приложите ваш файл с реализацией функции для автоматического тестирования

P.s. Для успешного выполнения задания необходимо поместить код решения в /home/user/string\_to\_int.go

### **Итоговое оценивание**

Итоговая аттестация по программе проводится на основе выполнения практических заданий, предусмотренных в рамках каждого тематического блока.

В ходе занятий слушатели последовательно осваивают синтаксис и инструменты языка Go, закрепляя материал через практику. Успешное выполнение всех заданий служит подтверждением освоения ключевых знаний, умений и навыков, предусмотренных программой.

Формат итоговой аттестации обеспечивает прозрачную, поэтапную оценку прогресса и позволяет объективно зафиксировать готовность слушателя к самостоятельной разработке на языке Go.